Reports and metrics can be easily found in the JuPyter notebooks shared. However there are few learnings and observations that can be taken away from the experiment.

Architectural Observations

1. Installing or setting up TensorRT can get a bit complicated if there's no local NVIDIA GPU or the machine is Windows based.
2. Even if we have a container, it still asks for a GPU or can only be run without it using NVIDIA Container Runtime which is only supported on a Linux based machine.
3. VMs or AMIs are quite costly for setting things up and running inference.
4. Configuring on Google Colab is an easy way out for those who don't have a local NVIDIA GPU and/or a Linux/Mac based machine.
5. CUDA version of Google Colab is 10.1 and the latest version of TensorRT that supports CUDA v10.1 is TensorRT 5.1.5.
6. TensorRT C++ APIs couldn't be used while working on Google Colab.

Procedural Observations

1. CIFAR10 is one of the benchmark datasets hence training a simple pretrained ResNet50 model on this was easy (with an accuracy of almost 91% in just one cycle).
2. The PyTorch model was converted to ONNX format but still required optimization and hence a simple command line utility came in handy for doing this (please see ../references.txt)

Performance Observations

1. The PyTorch model performed well on the test set but on running inference for a randomly sampled batch of test images, the engine performed extra-ordinarily well.
2. There was average drop of 1.2s per batch in the latency.
3. The model accuracy jumped from almost 91% to 95% on a single batch of test set.